LAT REQUIREMENTS DOCUMENT	Document # **LAT-TD-03075**	Date February 17, 2004
	Author(s) Richard Claus	Supersedes
	Subsystem/Office I&T	
Document Title **GLAST LAT I&T Online Roadmap to the July 2004 Freeze**		

This copy dated: 2/23/2004 10:11 AM

DRAFT

Gamma-ray Large Area Space Telescope (GLAST)

Large Area Telescope (LAT)

Integration and Test Subsystem (I&T)

Online Roadmap to the July 2004 Freeze

CHANGE HISTORY LOG

Revision	Effective Date	Description of Changes
01	February 17, 2004	Started initial version
02	February 23, 2004	Initial rough draft released after first review with I&T and Dick Horn. Fixed Table of Contents and added Power-up Sequence.

TABLE OF CONTENTS

1. Purpose 4
2. Scope 4
3. Definitions 4
3.1. Acronyms 4
3.2. Definitions 5
4. Applicable Documents 5
5. Description 6
6. The Roadmap 6
6.1. GASU support 6
6.1.1. Trigger interface 6
6.1.2. ACD script migration 6
6.1.3. EBF package update 6
6.2. Hardware monitoring system 7
6.3. Power-up sequence 7
6.4. Event prescaling in VxWorks 8
6.5. Subsystem scripts 8
6.6. Security against unauthorized software changes 8
6.7. Security against operator complacency 9
6.8. Operator training 9
6.9. Building 33 EGSE room 102 set-up 9
6.10. Release mechanism for subsystem test scripts to I&T 9
6.11. Electronic logbook 9
6.12. OPUS and the paper trail 9
6.13. Validation and Verification 9
6.14. Documentation 10
7. Bonus work 10
7.1. Test suites 10
7.2. Event display 10
7.3. Event batching 10
8. Background work 11
8.1. ELX support 11
8.2. FSW support 11
8.3. ACD, CAL and TKR support 11

1.

Purpose

The purpose of this document is to outline the roadmap of the Online department toward the start of LAT integration currently envisioned for July 2004.

2. Scope

The scope of this document covers Online department plans.

3. Definitions

3.1. Acronyms

ACD	Anti-coincidence subsystem
AEM	ACD Electronics Module
API	Application Program Interface
CAL	Calorimeter subsystem
COMM	Communications Module
cPCI	Compact PCI
EEPROM	Electrically Erasable Programmable Read Only Memory
EGSE	Electronic Ground Support Equipment
ELX	Electronics subsystem
EM	Engineering Model
EPU	Event Processor Unit
FITS	Flexible Image Transport System
FSW	Flight software
FU	Flight Unit
GUI	Graphical User Interface
I&T	Integration and Test
IDL	Interactive Data Language
ISOC	Instrument Science Operations Center
LAN	Local Area Network
LAT	Large Area Telescope
LATTE	LAT Test Executive
LCB	LAT Communications Board
MB	Mega bytes
NFS	Network File System
ODBC	Open DataBase Connectivity

PCI	Peripheral Communications Interface
PPC	Power PC
PSA	Power Supply Assembly
QU	Qualification Unit
RAM	Random Access Memory
ROOT	Rene's (?) Object Oriented Tool
RTE	Real Time Engine
SBC	Single Board Computer
SCL	Spacecraft Control Language
TBR	To Be Reviewed
TKR	Tracker subystem
TCP/IP	Transmission Control Protocol/Internet Protocol
TEM	Tower Electronics Module
VME	Versa Module Eurocard

3.2. Definitions

Flash Memory	A type of memory that retains its contents when powered off. It is normally read only but can be written by a special software procedure.
Functional test	A test that produces a go/no-go result.
Calibration procedure	A procedure whose product is a set of constants.
Subsystem	One of ACD, CAL, TKR, ELX or FSW
System	Something that crosses subsystem boundaries
Test-stand	The combination of a workstation, embedded system, electronics and sensor under test, with corresponding software
Single Contributor	A single contributor system, includes integrated TKR, CAL, TEM, PSA
Multi-contributor	Can include any subset of entire LAT system

4. Applicable Documents

- LAT-MD-00408 LAT Program Instrument Performance Verification Plan
- LAT-TD-00191 Tracker Tower Electrical Tests
- LAT-SS-00231 Calorimeter Performance Acceptance Standard and Tests
- LAT-TD-01112 ACD Functional Test Plans
- LAT-TD-00786 LAT Flight Software Test Plan
- LAT-MD-00676 LAT Assembly Sequence
- LAT-MD-00466 LAT I&T SVAC Plan

- LAT-MD-00440 LAT Beam Test Plan
- LAT-MD-01055 LAT Electrical Performance Test Plan
- LAT-SS-00XXX-XX LAT Electrical Ground Support Equipment Level 3 Specification
- LAT-SS-00586 Online Software Validation and Verification
- LAT-TD-02834 GLAT LAT I&T Testing Requirements Document

5. Description

This document describes the roadmap for the Online department toward the start of LAT integration in July 2004. It is in response to a request from Deputy for Systems manager Dick Horn. July 2004 is the time frame in which Flight Quality Parts are due to arrive in the Integration and Test building at SLAC (Building 33). This roadmap describes the steps necessary to ensure that releases of the LAT Test Executive (*LATTE*) and subsystem test scripts will be ready for use by the Integration and Test team with Flight Parts. LATTE and the test scripts are expected to remain as constant as possible throughout the integration period. Changes to the software will be permitted only through a Change Control Board –like protocol.

6. The Roadmap

6.1. GASU support

6.1.1. Trigger interface

The trigger interface API is described in "LAT-TD-???? GASU based teststands and the ACD v1.0 - First public draft". 1/3 of the Online group's resources are fully devoted to implementing this API, migrating LATTE to its use and testing it. As of February 18, 2004, much of the implementation work is currently complete. LATTE migration and testing is still to be done.

6.1.2. ACD script migration

As part of the testing of the Trigger interface, we plan to convert a version of the ACD scripts to the new triggering technique, supplied by the ACD EGSE group. This process will also provide examples for the ACD group to work from for future scripts.

6.1.3. EBF package update

The EBF package is used to parse event data, typically read from data files or the LATTE event data stream. This package is exported to SAS and others for use in reading data files for analysis. With the arrival of the GASU, the event format will change due to the addition of a contribution to the event from the GEM (trigger information).

Since both GASU-based teststands and mini-GLT based teststands will exist at the same time, the EBF package has to support event data formats generated by both types of teststand. In addition, the EBF package must support prior event data formats so that already existing event data files can continue to be processed.

The EBF package is provided in two forms: The first is as a library against which C++ code is built and the second is as a library against which Python code is run. The Python interface is

implemented using a third party package called SIP. SIP has recently undergone an API change. The old API is still supported but will go away at some point.

As of February 18, 2004, much of the implementation of the new format support is complete. What remains to be done is the support of the previous event format and implementation of the Python interface. A choice must be made whether to upgrade to the new SIP API or stay with the old one, realizing that the migration to the new interface will have to happen at some point. An analysis must still be made to determine how difficult the migration will be.

6.2. Hardware monitoring system

In LATTE releases up until now, the hardware monitoring system is tightly coupled with the Run Control system. In this situation, the system issues register commands to the hardware via a commanding client/server. This impacts system performance, both CPU-wise as well as network bandwidth-wise. Additionally, whenever the Run Control system isn't running, the hardware isn't being monitored. This is not acceptable when working with flight parts.

To make a system that constantly monitors the hardware whenever it is powered up, a process must be running on the host computer whenever it is powered up. This process must be started when the host boots. The system must be independent of Run Control and other processes running on the host. To achieve this, a *housekeeping server* running on the VxWorks SBC and a corresponding *housekeeping client* running on the host computer is envisaged.

The FSW group has described a design for a housekeeping system in a document called LAT Housekeeping (no LAT-TD-XXXXX assigned). To meet the July deadline, I&T will need to have (perhaps a degenerate portion, e.g., 1 tower, of) the system implemented in the April to May timeframe in order to meet the V&V phase of the plan. The Online department expects that the FSW group will supply the server portion of the system and the appropriate packet decoding tools. The Online department will provide the client side that allows:

- conversion to engineering units
- trending
- logging
- limit checking
- alarm handling
- alert generation

The client will repeatedly try to establish a connection to the server whenever one doesn't exist. Whenever the VxWorks crate is powered off, the client will be polling for a connection. As soon as VxWorks boots, the server will start and the connection will be established. At that point the hardware will be monitored.

6.3. Power-up sequence

The Power-up sequence is controlled by the PDU. Flight Software will provide the software to interact with the PDU upon a start-sequence command from LATTE. LATTE will report sequencing failures. The work involved on the LATTE side is not expected to be a big deal. However, the requirements on FSW need some investigation.

6.4. Event prescaling in VxWorks

LATTE, as a DAQ system, is performance limited by network bandwidth constraints. The LAT instrument is required handle an average L1 trigger rate of 10 KHz. This rate is greatly reduced to allow data transmission to the ground through on-board filtering. There is no requirement to sustain collection of a 10 KHz event rate from the LAT. None the less, an average 10 KHz trigger rate must be demonstrated and tested to ensure that the electronics responds correctly and within the tolerable error rate. Without having the completed flight software available, this can be achieved by not passing every event to LATTE.

Currently, the LATTE system supports the ability to transmit all the data from the VxWorks SBC or only the data containing errors, to the Run Control system. In the first case, deadtime will be introduced (trigger rate reduced) when the system saturates. In the second case, the trigger rate is allowed to increase arbitrarily providing that the error events don't saturate the system. The disadvantage of this second case is that it doesn't allow for monitoring the quality of the data passing through the system. It is therefore prudent to introduce a way of prescaling the transmitted events so that spot checking is possible.

Another method to improve event data throughput is to implement event batching, described below in section 7.3.

6.5. Subsystem scripts

Subsystems are expected to provide their scripts to the Online group prior to the V&V phase of the plan. At least a couple of months are required to complete V&V, so this implies that subsystem scripts are ready in the April time frame. Scripts are expected from:

- ELX
- CAL
- TKR
- ACD
- I&T

The scripts are outlined in "LAT-TD-02834, GLAT LAT I&T Testing Requirements Document". The I&T scripts are to be provided by the Online department. These constitute the "system" tests that cover inter-subsystem interactions.

Prior to release, the Online department will ensure that the scripts conform to I&T rules so that uniformity of the operator interface is maintained, the correct output files are produced, etc.

6.6. Security against unauthorized software changes

The Online department will implement techniques for ensuring that no deviances from released software can be run against flight hardware. LATTE currently recognizes when changes to the core software have been made and records that information in the run reports generated with each test. This will be extended to cover the released scripts. In addition, a command line option to Run Control will provide the ability to disable features of the system, such as loading arbitrary scripts and bringing up some GUIs. Permissions will be set up on the testing computer such that operators log into a captive account in which they will not be able to modify any files, including the batch file that launches Run Control.

Making this entirely bullet proof may be a difficult task, especially in a culture in which this sort of thing is like presenting a red flag in front a bull. However, detecting and recording that a breach has is not that difficult and is largely in place already.

6.7. Security against operator complacency

Operators will be required to sign off on any inputs they provide to the system by giving their private password. This still needs to be implemented.

6.8. Operator training

Once the system has more or less settled to its final form, an operator training program will be set up. This is to avoid key steps from being omitted, as happened during EM-1.

6.9. Building 33 EGSE room 102 set-up

We plan to set up the EGSE room in building 33 with two computers and four LCD screens. One computer will be the machine from which the tests are carried out. It will run Run Control and the test scripts. Its two screens will display the Run Control GUI, the HippoDraw GUI for visualizing data collected by the scripts and the electronic logbook. The other machine will be an observer computer that receives data from the data taking machine for opportunistic display, e.g. GOSED, the single event display. It will also be used to display the hardware monitoring system data.

Details of this still need to be worked out.

6.10. Release mechanism for subsystem test scripts to I&T

The subsystem scripts will be configuration controlled with cvs and released to the IFCT department computers for I&T testing by trained operators. The configuration control system and release process are already in place since similar techniques are used for LATTE itself.

6.11. Electronic logbook

An electronic logbook has been developed that uses the SLAC Oracle database. It is web-based and was used during the EM-1. A standalone version based on a MySQL database and Online's GUI tool Qt is currently being implemented. This allows electronic logbook use without there having to be a network connection to the central database. The standalone logbook works together with Run Control to be able to display the results from tests together with operator comments in real time. At any time later, the contents of the database can be exported to a file and then imported into the central database, thus providing the complete record of testing activity back at the ranch.

6.12. OPUS and the paper trail

OPUS is a tool provided by the SAS group for automating tasks. The Online department intends to use it for moving run-associated files off of the staging area on the testing computer to the official archive and for loading run information into the central log book. SAS and/or SVAC will use it to process the test data.

6.13. Validation and Verification

The validation and verification process will require a fair bit of time to explore the various situations that LATTE may find itself in. During EM-0 we spent 6 weeks to 2 months discovering all sorts of

issues with both hardware and software. The system has grown since then, but so has our familiarity with it. In addition, there are currently many more users of the system that help us shake the system out. Therefore, a rough guess at the time to do the V&V process is two to three months.

6.14. Documentation

Documentation has been lacking due to the Online departments efforts to provide subsystems with the features they need to do their jobs. We have redoubled our efforts to add to the documentation by inserting Doxygen comments into the code (describes the API) and adding to the Online System User's Guide (no LAT number assigned, yet). Much more work is to be done.

7. Bonus work

This section describes work that would be nice to get into the Flight Qualified release.

7.1. Test suites

A test suite is a set of test scripts or other suites that can be run under the control of LATTE. A suite invokes scripts or other suites sequentially. Given that suites can invoke other suites, the concept of nested scripts is provided.

The implementation of test suites was done as an afterthought and in a hurry to support CAL. This code really needs to be revisited to verify that it meets all the requirements and to improve its implementation. The Finite State Machine should be used to sequence the steps of each individual test in the suite. Functionality must be provided to allow aborting of any or all of the tests in the suite.

7.2. Event display

The event display tool the Online department has available now (GOSED) displays event data for only one tower. Diagnostic and error data from the tower contributions are not yet provided. The ACD is only beginning to be supported. The GEM (trigger) contribution is currently not yet displayable, mostly because of section 6.1.3 above not being completed. It is quite likely that some form of ACD display will be ready by July.

7.3. Event batching

Batching up events into larger packets will make those packets more efficiently transmittable over the Local Area Network between the VxWorks SBC and the host computer. This would allow higher rates of event data to be logged, although not an average rate of 10 KHz of realistically sized events (10 KHz * 2 Kbytes/event * 8 bits/byte = 160 Mbits/sec which is greater than the 100 Mbit/sec network links we have available. Gigabit Ethernet might be an option but would cost much more work. Secondly there are CPU constraints and disk write rates to consider.). Implementing a batching algorithm should not be too terribly difficult (shoulda done it to start with), but may have some repercussions downstream (the Run Control system). The EBF package to parse data already supports it. Since batching is not a requirement for hardware check-out, this item is listed as a bonus.

8. Background work

This section describes background work that must go on despite the preparation and maintenance of a Flight Qualified release. It may well mean that the Online group will be forced to prepare and maintain a separate release branch to support these activities. This is a condition that we would like to avoid due to the confusion that is likely to result. Maintaining multiple releases is nasty business.

8.1. ELX support

The Electronics group will be releasing new hardware in the coming months. The Online department will need to support interface additions so that the parts can be tested.

8.2. FSW support

The Flight Software group will be releasing new software in the coming months. They will also be informing the Online group of format and protocol changes that LATTE will need to support. As well, as we move closer to the flight interface, LATTE will need to be able to communicate with the SIU through the SIIS. This work is by and large orthogonal to the preparation for testing in Building 33, but will take time from the Online department's resources.

8.3. ACD, CAL and TKR support

ACD, CAL and TKR will continue to demand bug fixes and features to be added to LATTE until they get into flight hardware production. The Online department will need to support these requests.